CSE107: Intro to Modern Cryptography

https://cseweb.ucsd.edu/classes/sp22/cse107-a/

Emmanuel Thomé

March 31, 2022

Lecture 2

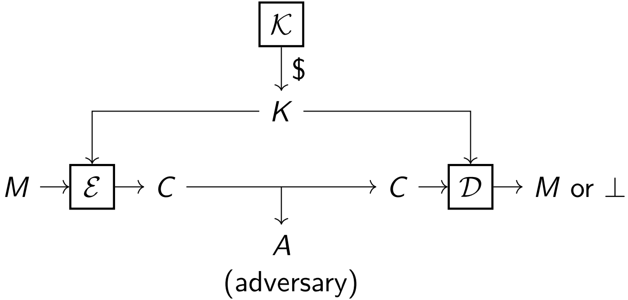
Classical Encryption

Examples

Perfect security

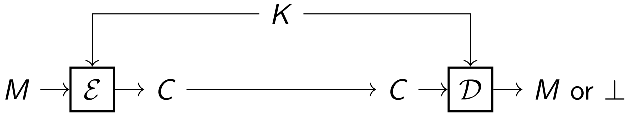
Syntax

A symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of three algorithms:



- $\mathcal{K}$ is the key generation algorithm.
- $\mathcal{E}$ is the encryption algorithm.
- $\mathcal{D}$ is the decryption algorithm.

Correct decryption requirement



For all K, M we have

$$\mathcal{D}_K(\mathcal{E}_K(M)) = M$$

Terminology recall

Alphabets:

- $\Sigma_1 = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \ldots, \mathtt{Z}\}$
- $\Sigma_2 = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \ldots, \mathtt{Z}\} \cup \{\sqcup, ., ?, \ldots\}$
- $\Sigma_3 = \{\mathtt{0}, \mathtt{1}\}$

Strings:

- Over Σ_1: HELLO, BZYK, ...
- Over Σ_2: HOW $\sqcup$ ARE $\sqcup$ YOU?
- Over Σ_3: 01101

Denote by Σ^* the set of all strings over alphabet Σ:

- $\{\mathtt{A}, \mathtt{B}, \ldots, \mathtt{Z}\}^*$
- $\{\mathtt{0}, \mathtt{1}\}^*$
- The empty string, denoted ε, is always in Σ^*.

Length and size

If s is a string then $|s|$ is the number of symbols in it:

- $|\texttt{HELLO}| = 5$
- $|\texttt{HOW} \sqcup \texttt{ARE} \sqcup \texttt{YOU?}| =$

Length and size

If s is a string then $|s|$ is the number of symbols in it:

- $|\texttt{HELLO}| = 5$
- $|\texttt{HOW} \sqcup \texttt{ARE} \sqcup \texttt{YOU?}| = 12$

Length and size

If s is a string then $|s|$ is the number of symbols in it:

- $|\texttt{HELLO}| = 5$
- $|\texttt{HOW} \sqcup \texttt{ARE} \sqcup \texttt{YOU?}| = 12$
- $|01101| = 5$

We denote by $s[i]$ the i-th symbol of string s:

- $s[3] = L$ if $s = \texttt{HELLO}$
- $s[5] = A$ if $s = \texttt{HOW} \sqcup \texttt{ARE} \sqcup \texttt{YOU?}$
- $s[2] = 1$ if $s = 01101$

If S is a set then $|S|$ is its size:

- $|\{\texttt{A}, \texttt{B}, \ldots, \texttt{Z}\}| = 26$
- $|\{0,1\}^8| =$

Length and size

If s is a string then $|s|$ is the number of symbols in it:

- $|\texttt{HELLO}| = 5$
- $|\texttt{HOW} \sqcup \texttt{ARE} \sqcup \texttt{YOU?}| = 12$
- $|01101| = 5$

We denote by $s[i]$ the i-th symbol of string s:

- $s[3] = L$ if $s = \texttt{HELLO}$
- $s[5] = A$ if $s = \texttt{HOW} \sqcup \texttt{ARE} \sqcup \texttt{YOU?}$
- $s[2] = 1$ if $s = 01101$

If S is a set then $|S|$ is its size:

- $|\{\texttt{A}, \texttt{B}, \ldots, \texttt{Z}\}| = 26$
- $|\{0, 1\}^8| = 2^8 = 256$

Functions

Notation: functions

Then notation $\pi : D \to R$ means π is a map (function) with

- inputs drawn from the set D (the domain)
- outputs falling in the set R (the range)

Example: Define $\pi : \{1, 4, 6\} \to \{0, 1\}$ by

x	1	4	6
$\pi(x)$	1	1	0

Functions can be specified as above or sometimes by code.

Example: The above can also be specified by

Alg $\pi(x)$
Return x mod 3

Permutations

Definition: permutation

A map (function) $\pi : S \to S$ is a permutation if it is one-to-one. Equivalently, it has an inverse map $\pi^{-1} : S \to S$.

Example: $S = \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$

A permutation and its inverse:

x	A	B	C
$\pi(x)$	C	A	B

y	A	B	C
$\pi^{-1}(x)$	B	C	A

Not a permutation:

x	A	B	C
$\pi(x)$	C	B	B

Counting permutations

There are many different possible permutations π: $S \to S$ on a given set S. How many?

To be specific: How many permutations $\pi : S \to S$ are there on the set $S = \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$?

Counting permutations

There are many different possible permutations π: $S \to S$ on a given set S. How many?

To be specific: How many permutations $\pi : S \to S$ are there on the set $S = \{\texttt{A}, \texttt{B}, \texttt{C}\}$?

Answer: $3! = 3 * 2 * 1 = 6$

x	$\pi(x)$	
A		$\longleftarrow$ 3 choices: A,B,C
B		$\longleftarrow$ 2 choices: not $\pi(\texttt{A})$
C		$\longleftarrow$ 1 choice: not $\pi(\texttt{A})$,$\pi(\texttt{B})$

In general there are $|S|!$ permutations $\pi : S \to S$.
Note that $n!$ is a fast-growing function: $n!$ has roughly $n \log n$ bits.

We let $\mathrm{Perm}(S)$ denote the set of all these permutations.

Plan

Examples

Perfect security

Substitution ciphers

- Alphabet Σ
- Key is a permutation $\pi : \Sigma \rightarrow \Sigma$ defining the encoding rule
- Plaintext $M \in \Sigma^*$ is a string over Σ
- Encryption of $M = M[1] \cdots M[n]$ is

$$C = \pi(M[1]) \cdots \pi(M[n])$$

- Decryption of $C = C[1] \cdots C[n]$ is

$$M = \pi^{-1}(C[1]) \cdots \pi^{-1}(C[n])$$

Substitution ciphers

Definition: substitution cipher

A substitution cipher over alphabet Σ is a symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in which the key output by $\mathcal{K}$ is a permutation $\pi : \Sigma \to \Sigma$, and

Algorithm $\mathcal{E}_\pi(M)$	Algorithm $\mathcal{D}_\pi(C)$
For $i = 1, \ldots, \lvert M \rvert$ do	For $i = 1, \ldots, \lvert C \rvert$ do
$C[i] \leftarrow \pi(M[i])$	$M[i] \leftarrow \pi^{-1}(C[i])$
Return C	Return M

Setup for Examples

$\Sigma = \{\mathtt{A}, \mathtt{B}, \ldots, \mathtt{Z}\} \cup \{\sqcup, ., ?, !, \ldots\}$

Plaintexts are members of Σ^*, which means any English text (sequence of sentences) is a plaintext.

For simplicity we only consider permutations that are punctuation respecting:

$$\pi(\sqcup) = \sqcup \quad , \quad \pi(.) = . \quad , \quad \pi(?) = ? \quad , \quad \ldots$$

so punctuation is left unchanged by encryption.

Example

σ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi(\sigma)$	B	U	P	W	I	Z	L	A	F	N	S	G	K
σ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi(\sigma)$	D	H	T	J	X	C	M	Y	O	V	E	Q	R

Then encryption of plaintext $M = \texttt{HI THERE}$ is

$$C \;=\; \pi(\texttt{H})\pi(\texttt{I})\pi(\sqcup)\pi(\texttt{T})\pi(\texttt{H})\pi(\texttt{E})\pi(\texttt{R})\pi(\texttt{E}) \;=\; \texttt{AF MAIXI}$$

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$	H	A	S	N	X	I	L	O	E	Q	M	G	T
τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$	J	V	C	Y	Z	K	P	B	W	D	R	U	F

Decryption of ciphertext $C = \texttt{AF MAIXI}$ is

$$\pi^{-1}(\texttt{A})\pi^{-1}(\texttt{F})\pi^{-1}(\sqcup)\pi^{-1}(\texttt{M})\pi^{-1}(\texttt{A})\pi^{-1}(\texttt{I})\pi^{-1}(\texttt{X})\pi^{-1}(\texttt{I}) \;=\; \texttt{HI THERE}$$

Cryptanalysis

Plaintext recovery

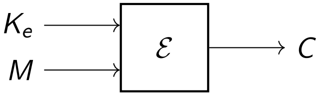
Basic adversary goal is plaintext recovery: given ciphertext C it aims to compute $M = \mathcal{D}(\pi, C)$.

This is easy if adversary knows π (hence π^{-1}), but adversary is not given the key π.

However it does know what encryption scheme is used. (Meaning, in this case, a substitution cipher.)

Note: in this class, we will define many other possible goals for the adversary.

Kerckhoffs's principle



Designers sometimes hope to get security by keeping the description of the encryption procedure $\mathcal{E}$ private. This is called security through obscurity.

But this prohibits standardization and usage.
And it tends not to add to security since adversaries are remarkably good at reverse engineering a description of $\mathcal{E}$ from any software or hardware artifact (executable program, encryption device, ...).
(Example: RC4 and "alleged-RC4".)

Kerckhoffs's principle (1883)

Good design (Kerckhoffs's principle):

- Adversary knows the system $\mathcal{E}$.
- The only thing it doesn't know is the key in use.

Cryptanalysis of a substitution cipher

Adversary has a ciphertext

```
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX
PTI.
```

Exploit structure of English: In typical text

- E is the most common letter
- Next are T, A, O, I, N, S, H, R

A letter by itself (like T in ciphertext) can only be A or I.

Etc.

Frequency counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3												

N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Frequency counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3											

N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Frequency counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	8	3	2	4	0	8	3	4	0	13	0	0

Cryptanalysis

```
  E E   E        E          '     E   E   E
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

      E     E:   E   E    '           ,       E
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

 E    .
OX PTI.
```

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	8	3	2	4	0	8	3	4	0	13	0	0

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$													

τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$											E		

Cryptanalysis

```
 E E  E         E          '      E    E    E
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

        E      E:   E    E     '
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

 E     .
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$													

τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

OX in ciphertext $\Rightarrow \pi^{-1}(\texttt{O}) \in \{\texttt{B},\texttt{H},\texttt{M},\texttt{W}\}$

Guess $\pi^{-1}(\texttt{O}) = \texttt{H}$ since O has pretty high frequency

Cryptanalysis

```
 HE E   E       E           '     E  HE  HE  H
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

      E     E:  HE  HE    '                 ,      HE
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE    .
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$													

τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

Cryptanalysis

```
 HE E   E        E          '     E  HE  HE  H
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

      E     E:  HE  HE    '           ,      HE
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE    .
OX PTI.
```

```
*HE*E
COXBX
```

Could be: THERE,THESE,WHERE,...

Guess $\pi^{-1}(C) = T$ since there is no ? in ciphertext so WHERE is unlikely.

So $\pi^{-1}(\mathtt{B}) \in \{\mathtt{R},\mathtt{S}\}$

Cryptanalysis

```
THE E   E T   T  E          '     E  HE  HE  H
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

  T   E    TE:  HE  HE    'T         T,      HE
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE    .
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$			T										

τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

Cryptanalysis

```
THE E   E T   T  E            '     E  HE  HE  H
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

  T   E     TE:  HE  HE    'T          T,      HE
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE    .
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$			T										

τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H									E		

T is a single-letter word so $\pi^{-1}(T) \in \{\mathtt{A}, \mathtt{I}\}$
We know $\pi^{-1}(\mathtt{B}) \in \{\mathtt{R}, \mathtt{S}\}$
So TBX could be: ARE,ASE,IRE,ISE
We guess ARE

Cryptanalysis

```
THERE ARE T   T  E     A  A ’     E  HE  HE  H
COXBX TBX CVK CDGXR DI T GTI’R ADHX VOXI OX ROKQAU

  T   E   ATE:  HE  HE  A ’T A   R   T, A    HE
IKC RNXPQATCX: VOXI OX PTI’C THHKBU DC, TIU VOXI

HE  A .
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$		R	T										
τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H					A				E		

Cryptanalysis

```
THERE ARE T   T  E     A  A ’     E  HE  HE  H
COXBX TBX CVK CDGXR DI T GTI’R ADHX VOXI OX ROKQAU

  T   E   ATE:  HE  HE  A ’T A   R   T, A    HE
IKC RNXPQATCX: VOXI OX PTI’C THHKBU DC, TIU VOXI

HE  A .
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$		R	T										
τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$		H					A				E		

```
*T
DC
```

D must be: A or I but T is A so D is I.

Etc....!

Cryptanalysis

```
THERE ARE TWO TIMES IN A MAN'S LIFE WHEN HE SHOULD
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU

NOT SPECULATE: WHEN HE CAN'T AFFORD IT, AND WHEN
IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI

HE CAN.
OX PTI.
```

τ	A	B	C	D	E	F	G	H	I	J	K	L	M
$\pi^{-1}(\tau)$	L	R	T	I			M	F	N		O		
τ	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$\pi^{-1}(\tau)$	P	H	C	U	S		A	D	W		E		

Assessment of security of substitution ciphers

Defenders may argue

- Cryptanalysis requires long ciphertext
- Harder if π is not punctuation-respecting

In fact substitution ciphers or variations and enhancements have been almost universally used until relatively recently.

Yet they are fundamentally flawed.

Hydraulic Telegraph

(Ancient Greece, 3rd and 4th century BC; link)

Messages written at prescribed heights on a rod.

To send a message:

1. Signal start using torch.
2. Open spigot.
3. When water level reaches desired message, close spigot.
4. Signal stop using torch.

Is this a secure encryption scheme?

Voting

Shall California adopt permanent Daylight Savings Time?

- YES/SI
- NO/NO

Voters V_1, V_2, V_3, V_4, V_5 cast votes at polling station.

Example votes: `YNYYN`

$$\text{Polling station} \xrightarrow{\pi(\texttt{Y})\pi(\texttt{N})\pi(\texttt{Y})\pi(\texttt{Y})\pi(\texttt{N})} \text{Tally center}$$

Is this secure?

Voting

Say $\pi(\mathtt{Y}) = \mathtt{A}$ and $\pi(\mathtt{N}) = \mathtt{B}$. Adversary sees

$$\pi(\mathtt{Y})\pi(\mathtt{N})\pi(\mathtt{Y})\pi(\mathtt{Y})\pi(\mathtt{N}) = \mathtt{ABAAB}$$

Adversary can infer relations: V_1, V_3 had same vote.

Adversary might be V_1

- It knows its own vote is Y
- So given ciphertext ABAAB it infers that A represents Y
- But then B must represent N
- Adversary knows everyone's vote!

The weakness

The weakness of a substitution cipher exploited above is simply that the same symbol is always encoded in the same way.

Attack does not require long plaintexts, and does not need π to be punctuation-respecting.

What happened?

Critical security thinking yielded a scenario where substitution ciphers fail miserably:

- Few possible plaintext symbols (`Y` or `N`)
- Adversary is one of the users (voters)

What did we learn?

- Security depends on usage
- Evaluating security requires being creative about coming up with usage scenarios that test the scheme

Good cryptography

A good scheme is one that

- Is secure in ALL (reasonable) scenarios
- Does not rely on obscurity. (i.e. encryption devices, or software, are known to the adversary)
- Is secure regardless of what type of data (e.g., `Y`,`N` strings) is being encrypted
- Even if adversary knows some decryptions, it shouldn't be able to produce others.

Plan

Examples

Perfect security

One time pad

Key $K \xleftarrow{\$} \{0,1\}^m$ is a random m-bit string

Plaintext $M \in \{0,1\}^m$ is an m-bit string

Algorithm $\mathcal{E}_K(M)$	Algorithm $\mathcal{D}_K(C)$
$C \leftarrow K \oplus M$	$M \leftarrow K \oplus C$
Return C	Return M

Assume only a single message M is ever encrypted under one key.

Voting

Represent `Y` by 1 and `N` by 0
Voters $V_1, \ldots, V_m$ cast votes $1, 0, 1, 1, 0, \ldots$

Let $M = 10110 \cdots$
Encryption is $C = K \oplus M$

Adversary has C but NOT K

Adversary cannot tell whether two people have same vote.

Even if adversary is V_1 and knows its own vote is 1, it cannot determine votes of other parties.

A measure of security

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. For any message M and ciphertext C we are interested in

$$\Pr\left[\mathcal{E}_K(M) = C\right]$$

where the probability is over the random choice $K \xleftarrow{\$} \mathcal{K}$ and over the coins tossed by $\mathcal{E}$ if any.

Example

Messages:

Keys:	00	01	10	11
00	01	10	11	00
01	01	11	10	00
10	00	11	01	11
11	11	10	01	11

The table entry in row K and column M is $\mathcal{E}_K(M)$.

- $\Pr[\mathcal{E}_K(00) = 01] =$

Example

Messages:

Keys:		00	01	10	11
	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row K and column M is $\mathcal{E}_K(M)$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] =$

Example

Messages:

Keys:	00	01	10	11
00	01	10	11	00
01	01	11	10	00
10	00	11	01	11
11	11	10	01	11

The table entry in row K and column M is $\mathcal{E}_K(M)$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$
- $\Pr[\mathcal{E}_K(10) = 11] =$

Example

Messages:

Keys:		00	01	10	11
	00	01	10	11	00
	01	01	11	10	00
	10	00	11	01	11
	11	11	10	01	11

The table entry in row K and column M is $\mathcal{E}_K(M)$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$
- $\Pr[\mathcal{E}_K(10) = 11] = \frac{1}{4}$

Perfect security

Definition: perfect security

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. We say that $\mathcal{SE}$ is perfectly secure if for any two messages $M_1, M_2 \in$ Plaintexts and any C

$$\Pr\left[\mathcal{E}_K(M_1) = C\right] \;=\; \Pr\left[\mathcal{E}_K(M_2) = C\right] \;.$$

The probability is over the random choice $K \xleftarrow{\$} \mathcal{K}$ and over the coins tossed by $\mathcal{E}$ if any.

Intuitively: Given C, and even knowing the message is either M_1 or M_2 the adversary cannot determine which.

Perfect security

Definition requires that

For all M_1, M_2, C we have

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[\mathcal{E}_K(M_2) = C] \; .$$

If we want to show the definition is not met, we need to show that

There exists M_1, M_2, C such that

$$\Pr[\mathcal{E}_K(M_1) = C] \neq \Pr[\mathcal{E}_K(M_2) = C] \; .$$

Example

Messages:

Keys:	00	01	10	11
00	01	10	11	00
01	01	11	10	00
10	00	11	01	11
11	11	10	01	11

The table entry in row K and column M is $\mathcal{E}_K(M)$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$

Is this encryption scheme perfectly secure?

Example

Messages:

Keys:	00	01	10	11
00	01	10	11	00
01	01	11	10	00
10	00	11	01	11
11	11	10	01	11

The table entry in row K and column M is $\mathcal{E}_K(M)$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{2}{4} = \frac{1}{2}$
- $\Pr[\mathcal{E}_K(01) = 01] = 0$

Is this encryption scheme perfectly secure?

No, because for $M_1 = 00$, $M_2 = 01$ and $C = 01$ we have

$$\underbrace{\Pr\left[\mathcal{E}_K(M_1) = C\right]}_{1/2} \neq \underbrace{\Pr\left[\mathcal{E}_K(M_2) = C\right]}_{0} .$$

(Im)Perfect security of substitution ciphers

A substitution cipher is NOT perfectly secure.

Formally:

Claim: Substitution is not perfectly secure

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a substitution cipher over the alphabet Σ consisting of the 26 English letters. Assume that $\mathcal{K}$ picks a random permutation over Σ as the key. That is, its code is

$$\pi \xleftarrow{\$} \mathsf{Perm}(\Sigma)\ ;\ \mathrm{return}\ \pi\ .$$

Let Plaintexts be the set of all three letter English words. Then $\mathcal{SE}$ is not perfectly secure.

Proof of claim

To show: there exist $M_1, M_2, C \in \Sigma^3$ such that

$$\Pr\left[\mathcal{E}_\pi(M_1) = C\right] \neq \Pr\left[\mathcal{E}_\pi(M_2) = C\right] \ .$$

We have replaced K with π because the key here is a permutation.

Proof of claim

To show: there exist $M_1, M_2, C \in \Sigma^3$ such that

$$\Pr\left[\mathcal{E}_\pi(M_1) = C\right] \neq \Pr\left[\mathcal{E}_\pi(M_2) = C\right] .$$

We have replaced K with π because the key here is a permutation.

Let

- $C = \texttt{XYY}$
- $M_1 = \texttt{FEE}$
- $M_2 = \texttt{FAR}$

Proof of claim

To show: there exist $M_1, M_2, C \in \Sigma^3$ such that

$$\Pr\left[\mathcal{E}_\pi(M_1) = C\right] \neq \Pr\left[\mathcal{E}_\pi(M_2) = C\right] .$$

We have replaced K with π because the key here is a permutation.

Let

- $C = \texttt{XYY}$
- $M_1 = \texttt{FEE}$
- $M_2 = \texttt{FAR}$

Then

$$\Pr\left[\mathcal{E}_\pi(M_2) = C\right] \quad = \quad \Pr\left[\pi(\texttt{F})\pi(\texttt{A})\pi(\texttt{R}) = \texttt{XYY}\right]$$

Proof of claim

To show: there exist $M_1, M_2, C \in \Sigma^3$ such that

$$\Pr\left[\mathcal{E}_\pi(M_1) = C\right] \neq \Pr\left[\mathcal{E}_\pi(M_2) = C\right] .$$

We have replaced K with π because the key here is a permutation.

Let

- $C = \texttt{XYY}$
- $M_1 = \texttt{FEE}$
- $M_2 = \texttt{FAR}$

Then

$$\begin{aligned} \Pr\left[\mathcal{E}_\pi(M_2) = C\right] &= \Pr\left[\pi(\texttt{F})\pi(\texttt{A})\pi(\texttt{R}) = \texttt{XYY}\right] \\ &= 0 \end{aligned}$$

Because $\pi(\texttt{A})$ cannot equal $\pi(\texttt{R})$

Proof of claim

$$\begin{aligned}
\Pr\left[\mathcal{E}_\pi(M_1) = C\right] &= \Pr\left[\mathcal{E}_\pi(\texttt{FEE}) = \texttt{XYY}\right] \\
&= \frac{|\{\,\pi \in \mathsf{Perm}(\Sigma) \;:\; \mathcal{E}_\pi(\texttt{FEE}) = \texttt{XYY}\,\}|}{|\,\mathsf{Perm}(\Sigma)|} \\
&= \frac{|\{\,\pi \in \mathsf{Perm}(\Sigma) \;:\; \pi(\texttt{F})\pi(\texttt{E})\pi(\texttt{E}) = \texttt{XYY}\,\}|}{|\,\mathsf{Perm}(\Sigma)|}
\end{aligned}$$

Proof of claim

$$
\begin{aligned}
\Pr[\mathcal{E}_\pi(M_1) = C] &= \Pr[\mathcal{E}_\pi(\texttt{FEE}) = \texttt{XYY}] \\
&= \frac{|\{\, \pi \in \mathrm{Perm}(\Sigma) \;:\; \mathcal{E}_\pi(\texttt{FEE}) = \texttt{XYY} \,\}|}{|\,\mathrm{Perm}(\Sigma)|} \\
&= \frac{|\{\, \pi \in \mathrm{Perm}(\Sigma) \;:\; \pi(\texttt{F})\pi(\texttt{E})\pi(\texttt{E}) = \texttt{XYY} \,\}|}{|\,\mathrm{Perm}(\Sigma)|} \\
&= \frac{24!}{26!} \\
&= \frac{1}{650}\,.
\end{aligned}
$$

Summary

Definition: perfect security

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. We say that $\mathcal{SE}$ is perfectly secure if for any two messages $M_1, M_2 \in$ Plaintexts and any C

$$\Pr\left[\mathcal{E}_K(M_1) = C\right] \;=\; \Pr\left[\mathcal{E}_K(M_2) = C\right] \;.$$

Claim: Substitution is not perfectly secure

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a substitution cipher over the alphabet Σ consisting of the 26 English letters. Assume that $\mathcal{K}$ picks a random permutation over Σ as the key. Let Plaintexts be the set of all three letter English words. Then $\mathcal{SE}$ is not perfectly secure.

We have proved the claim by presenting M_1, M_2, C such that

$$\Pr\left[\mathcal{E}_K(M_1) = C\right] \neq \Pr\left[\mathcal{E}_K(M_2) = C\right] \;.$$

Intuition for One-Time-Pad (OTP) security

Recall that One-Time-Pad encrypts M to $\mathcal{E}_K(M) = K \oplus M$.

Suppose adversary gets ciphertext $C = 101$ and knows the plaintext M is either $M_1 = 010$ or $M_2 = 001$. Can it tell which?

No, because $C = K \oplus M$ so

- $M = 010$ iff $K = 111$
- $M = 001$ iff $K = 100$

but K is equally likely to be 111 or 100 and adversary does not know K.

Perfect security of OTP

Claim: OTP is perfectly secure

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the OTP scheme with key-length $m \geq 1$. Then $\mathcal{SE}$ is perfectly secure.

Want to show that for any M_1, M_2, C

$$\Pr[\mathcal{E}_K(M_1) = C] = \Pr[\mathcal{E}_K(M_2) = C]$$

That is

$$\Pr[K \oplus M_1 = C] = \Pr[K \oplus M_2 = C]$$

when $K \stackrel{\$}{\leftarrow} \{0,1\}^m$.

Example: $m = 2$

Messages:

Keys:		00	01	10	11
	00	00	01	10	11
	01	01	00	11	10
	10	10	11	00	01
	11	11	10	01	00

The entry in row K, column M of the table is $\mathcal{E}_K(M) = K \oplus M$.

- $\Pr[\mathcal{E}_K(00) = 01] =$

Example: $m = 2$

		Messages:			
		00	01	10	11
Keys:	00	00	01	10	11
	01	01	00	11	10
	10	10	11	00	01
	11	11	10	01	00

The entry in row K, column M of the table is $\mathcal{E}_K(M) = K \oplus M$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{1}{4}$
- $\Pr[\mathcal{E}_K(10) = 01] =$

Example: $m = 2$

		Messages:			
		00	01	10	11
Keys:	00	00	01	10	11
	01	01	00	11	10
	10	10	11	00	01
	11	11	10	01	00

The entry in row K, column M of the table is $\mathcal{E}_K(M) = K \oplus M$.

- $\Pr[\mathcal{E}_K(00) = 01] = \frac{1}{4}$
- $\Pr[\mathcal{E}_K(10) = 01] = \frac{1}{4}$

Proof of claim

Probability for M_1

$$\Pr[\mathcal{E}_K(M_1) = C] \;=\; \Pr[K \oplus M_1 = C]$$

Proof of claim

Probability for M_1

$$\begin{aligned}
\Pr\left[\mathcal{E}_K(M_1) = C\right] &= \Pr\left[K \oplus M_1 = C\right] \\
&= \frac{|\{\, K \in \{0,1\}^m \,:\, K \oplus M_1 = C \,\}|}{|\{0,1\}^m|}
\end{aligned}$$

Proof of claim

Probability for M_1

$$\begin{aligned}
\Pr[\mathcal{E}_K(M_1) = C] &= \Pr[K \oplus M_1 = C] \\
&= \frac{|\{\, K \in \{0,1\}^m \,:\, K \oplus M_1 = C \,\}|}{|\{0,1\}^m|} \\
&= \frac{1}{2^m}\,.
\end{aligned}$$

Proof of claim

Same for M_2

$$\begin{aligned}\Pr\left[\mathcal{E}_K(M_2) = C\right] &= \Pr\left[K \oplus M_2 = C\right] \\ &= \frac{\left|\{\, K \in \{0,1\}^m \,:\, K \oplus M_2 = C \,\}\right|}{\left|\{0,1\}^m\right|} \\ &= \frac{1}{2^m}\,.\end{aligned}$$

In fact, OTP is the only encryption scheme that achieves Shannon's perfect security.

Perfect security: Plusses and Minuses

+	-
Very good privacy	Key needs to be as long as message

What next

We want schemes to securely encrypt

- arbitrary amounts of data
- with a single, short (e.g., 128 bit) key

This will be possible once we relax our goal from perfect to computational security.

Plan:

- Study the primitives we will use, namely block ciphers
- Understand and define computational security of block ciphers and encryption schemes
- Use (computationally secure) block ciphers to build (computationally secure) encryption schemes